New visual Servoing control strategies in tracking tasks using a PKM

[i]A. Traslosheros, [ii]L. Angel, [i]J. M. Sebastián, [iii]F. Roberti, [iii]R. Carelli and [i]R. Vaca
[i]DISAM, Universidad Politécnica de Madrid, Madrid, España, [ii] Facultad de Ingeniera Electrónica Universidad Pontificia Bolivariana Bucaramanga, Colombia, [iii]Instituto de Automática, Universidad Nacional de San Juan, San Juan, Argentina

1. Introduction

Vision allows a robotic system to obtain a lot of information on the surrounding environment to be used for motion planning and control. When the control is based on feedback of visual information is called Visual Servoing. Visual Servoing is a powerful tool which allows a robot to increase its interaction capabilities and tasks complexity. In this chapter we describe the architecture of the Robotenis system in order to design two different control strategies to carry out tracking tasks. Robotenis is an experimental stage that is formed of a parallel robot and vision equipment. The system was designed to test joint control and Visual Servoing algorithms and the main objective is to carry out tasks in three dimensions and dynamical environments. As a result the mechanical system is able to interact with objects which move close to $2m/s$. The general architecture of control strategies is composed by two intertwined control loops: The internal loop is faster and considers the information from the joins, its sample time is $0.5ms$. Second loop represents the visual Servoing system and it is an external loop to the first mentioned. The second loop represents the main study purpose, it is based in the prediction of the object velocity that is obtained from visual information and its sample time is $8.3ms$. The robot workspace analysis plays an important role in Visual Servoing tasks, by this analysis is possible to bound the movements that the robot is able to reach. In this article the robot jacobian is obtained by two methods. First method uses velocity vector-loop equations and the second is calculated from the time derivate of the kinematical model of the robot. First jacobian requires calculating angles from the kinematic model. Second jacobian instead, depends on physical parameters of the robot and can be calculated directly. Jacobians are calculated from two different kinematic models, the first one determines the angles each element of the robot. Fist jacobian is used in the graphic simulator of the system due to the information that can be obtained from it. Second jacobian is used to determine off-line the work space of the robot and it is used in the joint and visual controller of the robot (in real time). The work space of the robot is calculated from the condition number of the jacobian (this is a topic that is not studied in article). The dynamic model of the mechanical system is based on Lagrange multipliers, and it uses forearms and end effector platform of non-negligible inertias for the

development of control strategies. By means of obtaining the dynamic model, a nonlinear feed forward and a PD control is been applied to control the actuated joints. High requirements are required to the robot. Although requirements were taken into account in the design of the system, additional protection is added by means of a trajectory planner. the trajectory planner was specially designed to guarantee soft trajectories and protect the system from exceeding its Maximum capabilities. Stability analysis, system delays and saturation components has been taken into account and although we do not present real results, we present two cases: Static and dynamic. In previous works (Sebastián, et al. 2007) we present some results when the static case is considered.
The present chapter is organized as follows. After this introduction, a brief background is exposed. In the third section of this chapter several aspects in the kinematic model, robot jacobians, inverse dynamic and trajectory planner are described. The objective in this section is to describe the elements that are considered in the joint controller. In the fourth section the visual controller is described, a typical control law in visual Servoing is designed for the system: Position Based Visual Servoing. Two cases are described: static and dynamic. When the visual information is used to control a mechanical system, usually that information has to be filtered and estimated (position and velocity). In this section we analyze two critical aspects in the Visual Servoing area: the stability of the control law and the influence of the estimated errors of the visual information in the error of the system. Throughout this section, the error influence on the system behaviour is analyzed and bounded.

2. Background

Vision systems are becoming more and more frequently used in robotics applications. The visual information makes possible to know about the position and orientation of the objects that are presented in the scene and the description of the environment and this is achieved with a relative good precision. Although the above advantages, the integration of visual systems in dynamical works presents many topics which are not solved correctly yet. Thus many important investigation centers (Oda, Ito and Shibata 2009) (Kragic and I. 2005) are motivated to investigate about this field, such as in the Tokyo University ((Morikawa, et al. 2007), (Kaneko, et al. 2005) and (Senoo, Namiki and Ishikawa 2004)) where fast tracking (up to $6m/s$ and $58m/s^2$) strategies in visual servoing are developed. In order to study and implementing the different strategies of visual servoing, the computer vision group of the UPM (Polytechnic University of Madrid) decided to design the Robotenis vision-robot system. Robotenis system was designed in order to study and design visual servoing controllers and to carry out visual robot tasks, specially, those involved in tracking where dynamic environments are considered. The accomplishment of robotic tasks involving dynamical environments requires lightweight yet stiff structures, actuators allowing for high acceleration and high speed, fast sensor signal processing, and sophisticated control schemes which take into account the highly nonlinear robot dynamics. Motivated by the above reasons we proposed to design and built a high-speed parallel robot equipped with a vision system.

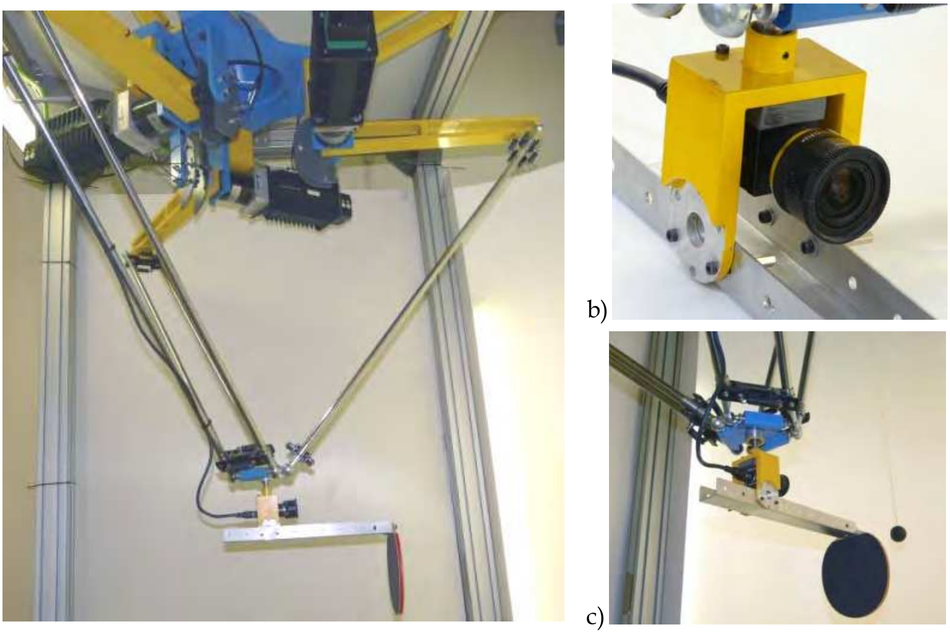


Fig. 1. Robotenis system and its environment: Robot, camera, background, ball and paddle.

The Robotenis System was created taking into account mainly two purposes. The first one is the development of a tool in order to use in visual servoing research. The second one is to evaluate the level of integration between a high-speed parallel manipulator and a vision system in applications with high temporary requirements. The mechanical structure of Robotenis System is inspired by the DELTA robot (Clavel 1988) (Stamper and Tsai 1997) and the vision system is based in one camera allocated at the end effector of the robot. The reasons that motivate us the choice of the robot is a consequence of the high requirements on the performance of the system, especially with regard to velocity, acceleration and the precision of the movements. The kinematic analysis and the optimal design of the Robotenis System have been presented by Angel, et al. (Angel, et al. 2005). The structure of the robot has been optimized from the view of both kinematics and dynamics respectively. The design method solved two difficulties: determining the dimensions of the parallel robot and selecting the actuators. In addition, the vision system and the control hardware was also selected.

3. Robotenis description

Basically, the Robotenis platform (Fig. 1.a) is formed by a parallel robot and a visual acquisition system. The parallel robot is based on a DELTA robot and its maximum end-effector speed is $4m=s$. The visual system is based on a camera in hand and its objective in this article resides in tracking a black ping pong ball. Visual control is designed by considering static and dynamic case. Static case considers that the desired distance between the camera and the ball is constant. Dynamic case considers that the desired distance between the ball and the camera can be changed at any time. Image processing is conveniently simplified using a black ball on white background. The ball is moved through a stick (Fig. 1.c) and the ball velocity is close to 2m=s. The visual system of the Robotenis platform is formed by a camera located at the end effector (Fig. 1.b) and a frame grabber (SONY XC-HR50 and Matrox Meteor 2-MC/4 respectively) The motion system is formed by AC brushless servomotors, Ac drivers (Unidrive) and gearbox.

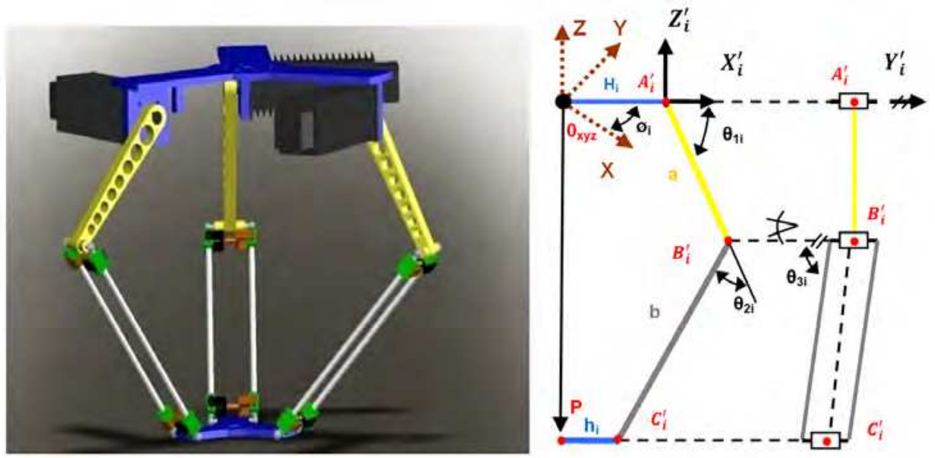


Fig. 2. Cad model and sketch of the robot that it is seen from the side of the i-arm
In section 3.1

3.1 Robotenis kinematical models

A parallel robot consists of a fixed platform that it is connected to an end effector platform by means of legs. These legs often are actuated by prismatic or rotating joints and they are connected to the platforms through passive joints that often are spherical or universal. In the Robotenis system the joints are actuated by rotating joints and connexions to end effector are by means of passive spherical joints. If we applied the Grüble criterion to the Robotenis robot, we could note that the robot has 9 DOF (this is due to the spherical joints and the chains configurations) but in fact the robot has 3 translational DOF and 6 passive DOF. Important differences with serial manipulators are that in parallel robots any two chains form a closed loop and that the actuators often are in the fixed platform. Above means that parallel robots have high structural stiffness since the end effector is supported in several points at the same time. Other important characteristic of this kind of robots is that they are able to reach high accelerations and forces, this is due to the position of the actuators in the fixed platform and that the end effector is not so heavy in comparison to serial robots. Although the above advantages, parallel robots have important drawbacks: the work space is generally reduced because of collisions between mechanical components and that singularities are not clear to identify. In singularities points the robot gains or losses degrees of freedom and is not possible to control. We will see that the Jacobian relates the actuators velocity with the end effector velocity and singularities occur when the Jacobian rank drops.

Nowadays there are excellent references to study in depth parallel robots, (Tsai 1999), (Merlet 2006) and recently (Bonev and Gosselin 2009).
For the position analysis of the robot of the Robotenis system two models are presented in order to obtain two different robot jacobians. As was introduced, the first jacobian is utilized in the Robotenis graphic simulator and second jacobian is utilized in real time tasks. Considers the Fig. 2, in this model we consider two reference systems. In the coordinate system Σ_{Oxyz} are represented the absolute coordinates of the system and the position "P" of the end effector of the robot. In the local coordinate system Σ'_{Oxyz} (allocated in each point A_i) the position and coordinates (A', B', C') of the i-arm are considered. The first kinematic model is calculated from Fig. 2 where the loop-closure equation for each limb is:

$$\overline{A'_i B'_i} + \overline{B'_i C'_i} = \overline{O_{xyz} P} + \overline{P C'_i} - \overline{O_{xyz} A'_i} \quad (1)$$

Expressing (note that $s(x) = sin(x)$ and $c(x) = cos(x)$ the eq. (1) in the coordinate system attached to each limb is possible to obtain:

$$\begin{bmatrix} C'_{ix} \\ C'_{iy} \\ C'_{iz} \end{bmatrix} = \begin{bmatrix} a\,c(\theta_{1i}) + b\,s(\theta_{3i})\,c(\theta_{1i}+\theta_{2i}) \\ b\,c(\theta_{3i}) \\ a\,s(\theta_{1i}) + b\,s(\theta_{3i})\,s(\theta_{1i}+\theta_{2i}) \end{bmatrix} \quad (2)$$

Where P and C_i are related by

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} c(\phi_i) & -s(\phi_i) & 0 \\ s(\phi_i) & c(\phi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \left[\begin{bmatrix} C'_{ix} \\ C'_{iy} \\ C'_{iz} \end{bmatrix} + \begin{bmatrix} -h_i + H_i \\ 0 \\ 0 \end{bmatrix} \right] \quad (3)$$

In order to calculate the inverse kinematics, from the second row in eq. (2), we have:

$$\theta_{3i} = c^{-1}\left(\frac{C'_{iy}}{b_i}\right) \quad (4)$$

θ_{2i} can be obtained by summing the squares of C_{ix}, C_{iy} and C_{iz} of the eq. (2).

$$C'^2_{ix} + C'^2_{iy} + C'^2_{iz} - a^2 - b^2 = 2ab\,s(\theta_{3i})\,c(\theta_{2i}) \quad \rightarrow \quad \theta_{2i} = -c^{-1}\left(\frac{C'^2_{ix} + C'^2_{iy} + C'^2_{iz} - a^2 - b^2}{2ab\,s(\theta_{3i})}\right) \quad (5)$$

By expanding left member of the first and third row of the eq. (2) by using trigonometric identities and making $\Psi_i = b_i \sin(\theta_{2i}) \sin(\theta_{3i})$ and $\Upsilon_i = a_i + b_i \cos(\theta_2) \sin(\theta_3)$:

$$\begin{aligned} \Upsilon_i c(\theta_{1i}) - \Psi_i s(\theta_{1i}) &= C'_{ix} \\ \Upsilon_i s(\theta_{1i}) - \Psi_i c(\theta_{1i}) &= C'_{iz} \end{aligned} \tag{6}$$

Note that from (6) we can obtain:

$$s(\theta_{1i}) = \left(\frac{\Upsilon_i C'_{iz} - \Psi_i C'_{ix}}{\Upsilon_i^2 + \Psi_i^2} \right) \quad \text{and} \quad c(\theta_{1i}) = \left(\frac{\Upsilon_i C'_{ix} + \Psi_i C'_{iz}}{\Upsilon_i^2 + \Psi_i^2} \right) \tag{7}$$

Equations in (7) can be related to obtain θ_{1i} as:

$$\theta_{1i} = \tan^{-1}\left(\frac{\Upsilon_i C'_{iz} - \Psi_i C'_{ix}}{\Upsilon_i C'_{ix} + \Psi_i C'_{iz}} \right) \tag{8}$$

In the use of above angles we have to consider that the "Z" axis that is attached to the center of the fixed platform it is negative in the space that the end effector of the robot will be operated. Taking into account the above consideration, angles are calculated as:

$$\theta_{3i} = -c^{-1}\left(\frac{C'_{iy}}{b} \right) \quad \theta_{2i} = -c^{-1}\left(\frac{C'^2_{ix} + C'^2_{iy} + C'^2_{iz} - a^2 - b^2}{2ab\,s(\theta_{3i})} \right) \quad \theta_{1i} = -\tan^{-1}\left(\frac{\Upsilon_i C'_{iz} - \Psi_i C'_{ix}}{\Upsilon_i C'_{ix} + \Psi_i C'_{iz}} \right) \tag{9}$$

Second kinematic model is obtained from Fig. 3.

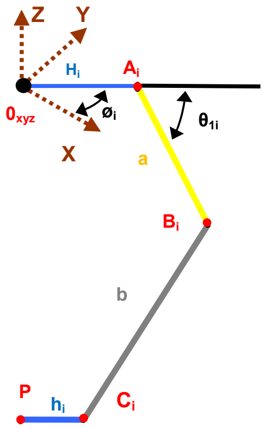


Fig. 3. Sketch of the robot taking into account an absolute coordinate reference system.

If we consider only one absolute coordinate system in Fig. 3, note that the segment $\overline{B_i C_i}$ is the radius of a sphere that has its center in the point B_i and its surface in the point C_i, (all points in the absolute coordinate system). Thus sphere equation as:

$$\Gamma_i = \left(C_{ix}-B_{ix}\right)^2+\left(C_{iy}-B_{iy}\right)^2+\left(C_{iz}-B_{iz}\right)^2-b^2=0 \tag{10}$$

From the Fig. 3 is possible to obtain the point $B_i = {}^{O_{xyz}}B_i$ in the absolute coordinate system.

$$^{O_{xyz}}\begin{bmatrix} B_{ix} \\ B_{iy} \\ B_{iz} \end{bmatrix} = \begin{bmatrix} \left(H_i + a\,\mathrm{c}(\theta_i)\right)\mathrm{c}(\phi_i) \\ \left(H_i + a\,\mathrm{c}(\theta_i)\right)\mathrm{s}(\phi_i) \\ a\,\mathrm{s}(\theta_i) \end{bmatrix} \qquad \text{where } \mu_i = \mu_{1i} \tag{11}$$

Replacing eq. (11) in eq. (10) and expanding it the constraint equation $\boldsymbol{\Gamma_i}$ is obtained:

$$\begin{aligned}\Gamma_i =& -H_i^2-a^2+b^2-C_{ix}{}^2-C_{iy}{}^2-C_{iz}{}^2+2H_iC_{ix}\mathrm{c}(\phi_i)+2H_iC_{iy}\mathrm{s}(\phi_i)-2H_ia\,\mathrm{c}(\theta_i)+2a\,C_{ix}\mathrm{c}(\theta_i)\mathrm{c}(\phi_i)\\ &-2a\,C_{iz}\mathrm{s}(\theta_i)+2a\,C_{iy}\mathrm{c}(\theta_i)\mathrm{s}(\phi_i)=0\end{aligned} \tag{12}$$

In order to simplify, above can be regrouped, thus for the i-limb:

$$E_i\,\mathrm{s}(\theta_i)+F_i\,\mathrm{c}(\theta_i)+M_i=0 \tag{13}$$

Where:

$$\begin{aligned}&F_i=2a\left(C_{ix}\mathrm{c}(\phi_i)+C_{iy}\mathrm{s}(\phi_i)-H_i\right) \qquad\qquad E_i=2a\,C_{iz}\\ &M_i=b^2-a^2-C_{ix}^2-C_{iy}^2-C_{iz}^2-H_i^2+2H_i\left(C_{ix}\mathrm{c}(\phi_i)+C_{iy}\mathrm{s}(\phi_i)\right)\end{aligned} \tag{14}$$

The following trigonometric identities can be replaced into eq. (13):

$$\mathrm{s}(\theta_i)\equiv\frac{2\tan\left(\frac{1}{2}\theta_i\right)}{1+\tan^2\left(\frac{1}{2}\theta_i\right)} \qquad \text{and} \qquad \mathrm{c}(\theta_i)\equiv\frac{1-\tan^2\left(\frac{1}{2}\theta_i\right)}{1+\tan^2\left(\frac{1}{2}\theta_i\right)} \tag{15}$$

And we can obtain the following second order equation:

$$\left(M_i-F_i\right)\tan^2\left(\frac{1}{2}\theta_i\right)+2E_i\tan\left(\frac{1}{2}\theta_i\right)+M_i+F_i=0 \tag{16}$$

And the angle θ_i can be finally obtained as:

$$\theta_i = 2\left[\tan^{-1}\left(\frac{-E_i \pm \sqrt{E_i^2 + \left(F_i^2 - M_i^2\right)}}{\left(M_i - F_i\right)}\right)\right] \tag{17}$$

Where C_{ix}, C_{iy} and C_{iz} are:

$$\begin{bmatrix} C_{ix} \\ C_{iy} \\ C_{iz} \end{bmatrix} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} - \begin{bmatrix} c(\phi_i) & -s(\phi_i) & 0 \\ s(\phi_i) & c(\phi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -h_i \\ 0 \\ 0 \end{bmatrix} \tag{18}$$

3.2 Robot Jacobians

In robotics, the robot Jacobian can be seen as the linear relation between the actuators velocity and the end effector velocity. In Fig. 4. direct and inverse Jacobian show them relation with the robot speeds. Although the jacobian can be obtained by other powerful methods (screws theory (Stramigioli and Bruyninckx 2001) (Davidson and Hunt Davidson 2004) or motor algebra (Corrochano and Kähler 2000)), conceptually the robot jacobian can be obtained as the derivate of the direct or the inverse kinematic model. In parallel robots the obtaining of the Jacobian by means of the screws theory or motor algebra can be more complicated. This complication is due to its non actuated joints (that they are not necessary passive joints). The easier method to understand, but not to carry out, is to derivate respect the time the kinematic model of the robot.

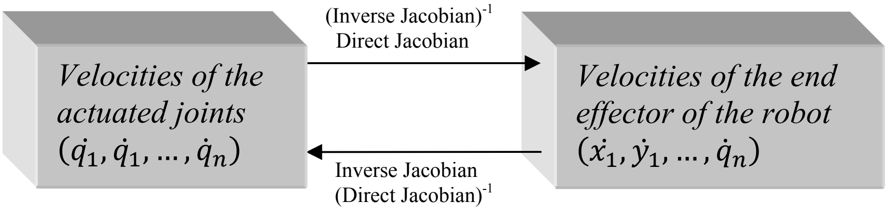


Fig. 4. Direct and indirect Jacobian and its relation with the robot velocities

Sometimes is more complex to obtain the inverse or direct Jacobian from one kinematic model than other thus, in some practical cases is possible to obtain the inverse Jacobian by inverting the direct Jacobian and vice versa, Fig. 4. Above proposal is easy to describe but does not analyze complications. For example if we would like to calculate the inverse Jacobian form the direct Jacobian, we have to find the inverse of a matrix that its dimension is normally 6 × 6. This matrix inversion it could be very difficult because components of the Jacobian are commonly complex functions and composed by trigonometric functions. Alternatively it is possible to calculate the inverse Jacobian or the direct Jacobian by other methods. For example if we have the algebraic form of the direct Jacobian, we could calculate the inverse of the Jacobian by means of inverting the numeric direct Jacobian (previously evaluated at one point). On the other hand the Jacobian gives us important

information about the robot, from the Jacobian we can determinate when a singularity occurs. There are different classifications for singularities of a robot. Some singularities can be classified according to the place in the space where they occurs (singularities can present on the limit or inside of the workspace of the robot). Another classification takes into account how singularities are produced (produced from the direct or inverse kinematics).
Suppose that the eq. (19) describes the kinematics restrictions that are imposed to mechanical elements (joints, arms, lengths, etc.) of the robot.

$$f(x,q)=0 \tag{19}$$

Where $\boldsymbol{x} \in \mathfrak{R}^{6\times1}$ is the position and orientation of the end effector of the robot and $\boldsymbol{q} \in \mathfrak{R}^{n\times1}$ are the joint variables that are actuated. Note that if $\mathbf{n} > 6$ the robot is redundant and if $\mathbf{n} < 6$ the robot cannot fully orientate ($\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$) or displace (along $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$) in the 3D space. Although sometimes a robot can be specially designed with other characteristics, in general a robot has the same number of DOF that its number of actuators.
Consider the time derivative of the eq. (19) in the following equation.

$$J_x\dot{x}=J_q\dot{q} \quad \text{where} \quad J_{\mathrm{x}}=\frac{\partial f(\mathrm{x},\mathrm{q})}{\partial \mathrm{x}} \quad \text{and} \quad J_{\mathrm{q}}=-\frac{\partial f(\mathrm{x},\mathrm{q})}{\partial \mathrm{q}} \tag{20}$$

Note that $\dot{x}$ and $\dot{q}$ are the time derivate of x and q respectely. The direct and the inverse Jacobian can be obtained as the following equations.

$$\dot{x}=J_D\dot{q} \quad \text{and} \quad \dot{q}=J_I\dot{x} \quad \text{where} \quad J_D=J_x^{-1}J_q \quad \text{and} \quad J_I=J_q^{-1}J_x \tag{21}$$

A robot singularity occurs when the determinant of the Jacobian is cero. Singularities can be divided in three groups: singularities that are due to the inverse kinematics, those that are due to the direct kinematics and those that occurs when both above singularities take place at the same time (combined singularities). For a non redundant robot (the Jacobian is a square matrix), each one of above singularities happens when: $Det(J_q) = 0$, $Det(J_x) = 0$ and when $Det(J_q) = Det(J_x) = 0$. Singularities can be interpreted differently in serial robots and in parallel robots. When we have that $Det(J_q) = 0$, it means that the null space of J_q is not empty. That is, there are values of $\dot{q}$ that are different from cero and produce an end effector velocity that is equal to cero $\dot{x} = 0$. In this case, the robot loses DOF because there are infinitesimal movements of the joints that do not produce movement of the end effector; commonly this occurs when robot links of a limb are in the same plane. Note that when an arm is completely extended, the end effector can supported high loads when the action of the load is in the same direction of the extended arm. On the other hand when $Det(J_x) = 0$ we have a direct kinematics singularity, this means that the null space of J_x is not empty. The above means that there are values of $\dot{x}$ that are different from cero when the actuators are blocked $\dot{q} = 0$. Physically, the end effector of the robot gains DOF. When the end effector gains DOF is possible to move infinitesimally although the actuators would be blocked. The third type of singularity it is a combined singularity and can occurs in parallel robots with special architecture or under especial considerations. Sometimes singularities can be identified from the Jacobian almost directly but sometimes Jacobian elements are really complex and singularities are difficult to identify. Singularities can be identified in easier manner depending on how the Jacobian is obtained. The Robotenis Jacobian is obtained by two methods, one it is obtained from the time derivate of a closure loop equation (1) and the second Jacobian is obtained from the time derivate of the second kinematic model. Remember that the jacobian obtained from the eq. (1) requires solving the kinematic model in eqs. (9). This jacobian requires more information of each element of the robot and it is used in the graphic simulator of the system. In order to obtain the jacobian consider that $O = O_{xyz}$ and that the eq. (1) is rearranged as:

$$\overline{O\,P} + R(\phi,z)\,\overline{P\,C'_i} = R(\phi_i,z)\left[\overline{O\,A'_i} + \overline{A'_iB'_i} - \overline{B'_iC'_i}\right] \tag{22}$$

Where $R(\emptyset, z)$ is a $\emptyset\ rad$ rotation matrix around the Z axis in the absolute coordinate system.

$$R(\phi,z) = \begin{bmatrix} c(\phi) & -s(\phi) & 0 \\ s(\phi) & c(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{23}$$

By obtaining the time derivate of the equation (22) and multiplying by $R^T(\emptyset, z)$ we have:

$$R^T(\phi_i,z)\,\dot{P} = \omega_{1i} \times \mathbf{a_i} + \omega_{2i} \times \mathbf{b_i} \tag{24}$$

Where $\dot{P}$ is velocity of the end effector in the XYZ coordinate system, $a_i = \overrightarrow{A'_iB'_i}$, $b_i = \overrightarrow{B'_iC'_i}$ and w_{1i}, w_{2i} are the angular velocities of the links 1 and 2 of the i limb. Observe that θ_{i2} and θ_{i3} are passive variables (they are not actuated) thus to eliminate the passive joint speeds (w_{2i}) we dot multiply both sides of eq. (24) by $\boldsymbol{b_i}$. By means of the properties of the triple product ($\vec{b} \cdot (\vec{\omega} \times \vec{a}) = \vec{\omega} \cdot (\vec{a} \times \vec{b})$ and $\vec{b} \cdot (\vec{\omega} \times \vec{b}) = \vec{\omega} \cdot (\vec{b} \times \vec{b}) = 0$) is possible to obtain:

$$b \cdot \left[R^T(\phi_i,z)\,\dot{P}\right] = \omega_{1i} \cdot (\mathbf{a}_i \times \mathbf{b}_i) \tag{25}$$

From Fig. 2 elements of above equation are:

$$\mathbf{a_i} = a\begin{bmatrix} c(\theta_{1i}) \\ 0 \\ s(\theta_{1i}) \end{bmatrix}; \quad \mathbf{b_i} = b\begin{bmatrix} s(\theta_{3i})\,c(\theta_{1i}+\theta_{2i}) \\ c(\theta_{3i}) \\ s(\theta_{3i})\,s(\theta_{1i}+\theta_{2i}) \end{bmatrix} \quad \text{and} \quad \omega_{1i} = \begin{bmatrix} 0 \\ -\dot{\theta}_{1i} \\ 0 \end{bmatrix} \tag{26}$$

All of them are expressed in the i −coordinate system. Substituting equations in (26) into (25) and after operating and simplifying we have:

$$\begin{bmatrix} m_{1x} & m_{1y} & m_{1z} \\ m_{2x} & m_{2y} & m_{2z} \\ m_{3x} & m_{3y} & m_{3z} \end{bmatrix} \begin{bmatrix} \dot{P}_x \\ \dot{P}_y \\ \dot{P}_z \end{bmatrix} = a \begin{bmatrix} s(\theta_{21})s(\theta_{31}) & 0 & 0 \\ 0 & s(\theta_{22})s(\theta_{32}) & 0 \\ 0 & 0 & s(\theta_{23})s(\theta_{33}) \end{bmatrix} \begin{bmatrix} \dot{\theta}_{11} \\ \dot{\theta}_{12} \\ \dot{\theta}_{13} \end{bmatrix} \tag{27}$$

Where

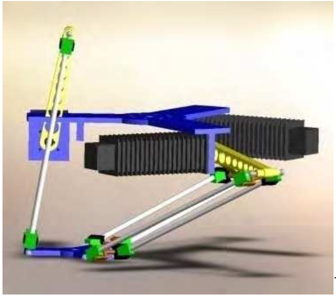
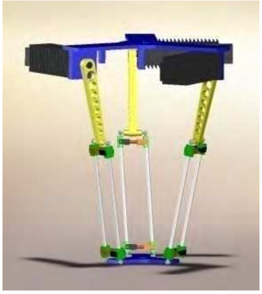
$$\begin{aligned} m_{ix} &= c(\theta_{1i}+\theta_{2i})s(\theta_{3i})c(\phi_i) - c(\theta_{3i})s(\phi_i) \\ m_{iy} &= c(\theta_{1i}+\theta_{2i})s(\theta_{3i})s(\phi_i) + c(\theta_{3i})c(\phi_i) \\ m_{iz} &= s(\theta_{1i}+\theta_{2i})s(\theta_{3i}) \end{aligned} \tag{28}$$

Note that the right and left part of the eq. (27) represents the inverse and direct Jacobians respectively. An inverse kinematic singularity occurs when $\theta_{2i} = 0 \; or \; \pi$ or $\theta_{3i} = 0 \; or \; \pi$, see Fig. 5 a) and b). On the other hand direct kinematic singularities occur when rows of the left matrix become linearly dependent. The above is:

$$k_1[m_1] + k_2[m_2] + k_3[m_3] = 0 \qquad \text{Where } k_1, k_2, k_3 \in \Re \text{ and not all are cero} \tag{29}$$

Equation (29) is not as clear as the right part of the equation (27) but we can identify a group of direct kinematic singularities when the last column in the three rows is cero, this is:

$$\begin{gathered} s(\theta_{11}+\theta_{21})s(\theta_{31}) = s(\theta_{12}+\theta_{22})s(\theta_{32}) = s(\theta_{13}+\theta_{23})s(\theta_{33}) = 0 \\ \text{When} \quad \theta_{1i}+\theta_{2i} = 0 \quad \text{or} \quad = \pi \quad \forall\, i=1,2,3 \qquad \text{or when} \quad \theta_{3i} = 0 \quad \text{or} \quad = \pi \quad \forall\, i=1,2,3 \end{gathered} \tag{30}$$

a)  b) 

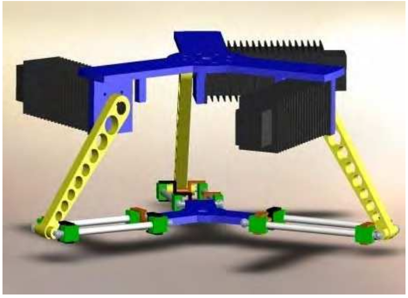
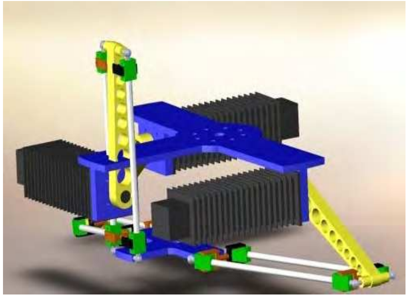
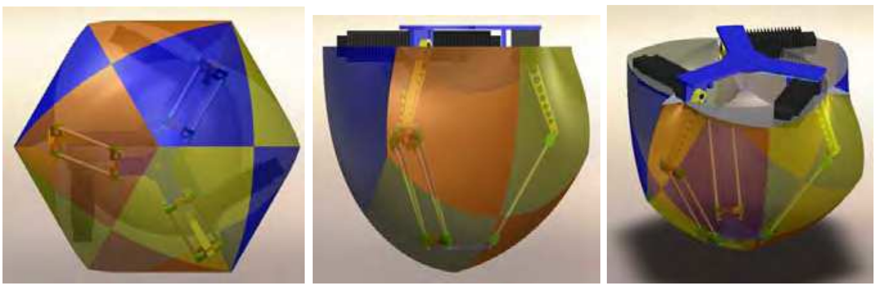
c)  d) 

Fig. 5. a) Inverse kinematic singularities if $\boldsymbol{\theta_{21}} = \boldsymbol{\pi}$. b) Inverse kinematic singularities where $\boldsymbol{\theta_{2i}} = \boldsymbol{0}$; $\forall\, \boldsymbol{i} = \boldsymbol{1}, \boldsymbol{2}, \boldsymbol{3}$. c) Direct kinematic singularity if $\boldsymbol{\theta_{1i}} + \boldsymbol{\theta_{2i}} = \boldsymbol{\pi}$; $\forall\, \boldsymbol{i} = \boldsymbol{1}, \boldsymbol{2}, \boldsymbol{3}$. d) Combined kinematic singularity if $\boldsymbol{\theta_{21}} = \boldsymbol{\pi}$; $\boldsymbol{\theta_{32}} = \boldsymbol{0}$; $\boldsymbol{\theta_{33}} = \boldsymbol{0}$; $\boldsymbol{\theta_{12}} + \boldsymbol{\theta_{22}} = \boldsymbol{\pi}$; $\boldsymbol{\theta_{13}} + \boldsymbol{\theta_{23}} = \boldsymbol{0}$ and $\boldsymbol{\phi_2}+\boldsymbol{\phi_3} = \boldsymbol{0}\ \boldsymbol{or}\ \boldsymbol{\pi}$. Note that the robot presents a combined singularity if three angles $\boldsymbol{\theta_{3i}} = \boldsymbol{\pi}$; $\forall\, \boldsymbol{i} = \boldsymbol{1}, \boldsymbol{2}, \boldsymbol{3}$ consequently case c) is a combined singularity ($\boldsymbol{\theta_{32}} = \boldsymbol{0}$; $\boldsymbol{\theta_{33}} = \boldsymbol{0}$). Note that the design of the robot plays a very important role because singularities can even avoid. For example in figure c) the singularity is present because lengths of the forearm allows to be in the same plane that the end effector platform and in the figure d) a combined singularity is present because $\boldsymbol{\phi_2}+\boldsymbol{\phi_3} = \boldsymbol{0}$. In all figures we suppose that the limb $\boldsymbol{i} = \boldsymbol{1}$ is the limb situated to the left of the images. Note that collisions between mechanical elements are not taken into account.

By considering (27) to (30), direct kinematic singularities present when the end effector platform is in the same plane as the parallelograms of the 3 limbs, in this configuration the robot cannot resist any load in the Z direction, see Fig. 5 c). Note that singularities like above depend on the lengths and angles of the robot when it was designed Fig. 5 c), such is the case of the above configuration where singularity can present when $\boldsymbol{a} + \boldsymbol{H} \geq \boldsymbol{b} +$, other singularities can present in special values of $\boldsymbol{\phi_i}$ Fig. 5 d).
Analysis of singularities of the work space is important for Visual controller in order to bound the workspace an avoid robot injures. Above analysis is useful because some singularities are given analytically. Different views of the work space of the CAD model of the Robotenis system is shown in Fig. 6



a) b) c)

Fig. 6 Work space of the Robotenis system. a) Work space is seen from bottom part of the robot, b) it shows the workspace from side. c) The isometric view of the robot is shown.

As was mentioned a second jacobian is obtained to use in real time tasks, by the condition number of the jacobian (Yoshikawa 1985) is possible to know how far a singularity is. Condition number of the jacobian y checked before carry out any movement of the robot, if a large condition number is present, then the movement is not carried out.
In order to obtain the second jacobian consider that we have the inverse kinematic model of a robot in given by eq. (31).

$$\begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix} = \begin{bmatrix} f_1(x, y, z, \alpha, \beta, \gamma) \\ \vdots \\ f_n(x, y, z, \alpha, \beta, \gamma) \end{bmatrix} \tag{31}$$

$$\begin{bmatrix} \dot{q}_1 \\ \vdots \\ \dot{q}_n \end{bmatrix} = J_I \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} \qquad \text{Where:} \qquad J_I = \begin{bmatrix} \frac{\partial f_1}{\partial x} & \cdots & \frac{\partial f_n}{\partial \gamma} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x} & \cdots & \frac{\partial f_n}{\partial \gamma} \end{bmatrix} \tag{32}$$

Note that the kinematic model of the Robotenis system is formed by three equations in eq. (17) (the end effector cannot orientate) and this model has the form of the eq. (31). Consequently to obtain the jacobian we have to find the time derivate of the kinematic model. Thus to simplify operations we suppose that.

$$\psi_i(t) = \frac{-E_i \pm \sqrt{E_i^2 + \left(F_i^2 - M_i^2\right)}}{\left(M_i - F_i\right)} \tag{33}$$

And that in terms of $\psi$the time derivate of (17) is:

$$\dot{\theta}_i = 2\left(\frac{\dot{\psi}_i}{1+\psi_i^2}\right) \tag{34}$$

Where $\dot{\psi}_i$ is

$$\dot{\psi}_i = \left[\frac{-\dot{E}}{M-F} + \frac{P\left(\dot{M}-\dot{F}\right)}{(M-F)^2} \pm \left[\frac{E\,\dot{E} - M\,\dot{M} + F\,\dot{F}}{(M-F)\sqrt{E^2 - \left(M^2 - F^2\right)}} + \frac{-\sqrt{E^2 - \left(M^2 - F^2\right)}\left(\dot{M}-\dot{F}\right)}{(M-F)^2}\right]\right]_i \tag{35}$$

Considering that $\eta_1 = \frac{1}{M-F}$ and $\eta_2 = \frac{1}{\sqrt{E^2 - (M^2 - F^2)}}$ can be replaced in (35).

$$\dot{\psi}_i = \left[-\eta_1 \dot{E} + \eta_1^2 E\,\dot{M} - \eta_1^2 E\,\dot{F} \pm \left[\eta_1 \eta_2 E\,\dot{E} - \eta_1 \eta_2 M\,\dot{M} + \eta_1 \eta_2 F\,\dot{F} - \frac{\eta_1^2 \dot{M}}{\eta_2} + \frac{\eta_1^2 \dot{F}}{\eta_2}\right]\right]_i \tag{36}$$

On the other hand we know that:

$$\begin{aligned}&\dot{F}_i=2a\dot{C}_{ix}\mathrm{c}(\phi_i)+2a\dot{C}_{iy}\mathrm{s}(\phi_i)\\&\dot{M}_i=-2C_{ix}\dot{C}_{ix}-2C_{iy}\dot{C}_{iy}-2C_{iz}\dot{C}_{iz}+2H\dot{C}_{ix}\mathrm{c}(\phi_i)+2H\dot{C}_{iy}\mathrm{s}(\phi_i)\\&\dot{E}_i=2a\dot{C}_{iz}\end{aligned}\tag{37}$$

By rearranging terms in eq. (36) and considering terms in (37) is possible to obtain ψ in terms of the velocity of the end effector $\dot{C}_{xyz}$.

$$\dot{\psi}_i=2\left(d_{ix}\dot{C}_{ix}+d_{iy}\dot{C}_{iy}+d_{iz}\dot{C}_{iz}\right)\tag{38}$$

Where:

$$d_{ix}=\begin{bmatrix}\eta_1^2P\left[-C_x+H\mathrm{c}(\phi)-a\mathrm{c}(\phi)\right]\pm\\ \begin{bmatrix}\eta_1\eta_2\left[MC_x+Fa\mathrm{c}(\phi)-MH\mathrm{c}(\phi)\right]+\\ \dfrac{\eta_1^2}{\eta_2}\left[a\mathrm{c}(\phi)+C_x-H\mathrm{c}(\phi)\right]\end{bmatrix}\end{bmatrix}_i,\quad d_{iy}=\begin{bmatrix}\eta_1^2P\left[-C_y+H\,s(\phi)-as(\phi)\right]\pm\\ \begin{bmatrix}\eta_1\eta_2\left[F\,as(\phi)+MC_y-MH\,s(\phi)\right]+\\ \dfrac{\eta_1^2}{\eta_2}\left[as(\phi)+C_y-H\,s(\phi)\right]\end{bmatrix}\end{bmatrix}_i,$$

$$d_{iz}=\eta_1\begin{bmatrix}-a-\eta_1EC_z\pm\\ \left[\eta_2\left(Pa+MC_z\right)+\dfrac{\eta_1}{\eta_2}C_z\right]\end{bmatrix}_i\tag{39}$$

Then replacing eq. (38) in (34) we have:

$$\dot{\theta}_i=4\frac{d_{ix}\dot{C}_{ix}+d_{iy}\dot{C}_{iy}+d_{iz}\dot{C}_{iz}}{1+\left(\dfrac{-E\pm\sqrt{E^2+\left(F^2-M^2\right)}}{2(M-F)}\right)_i^2}\quad\text{and}\quad\dot{\theta}_i=4\begin{bmatrix}D_{ix} & D_{iy} & D_{iz}\end{bmatrix}\begin{bmatrix}\dot{C}_{ix}\\ \dot{C}_{iy}\\ \dot{C}_{iz}\end{bmatrix}\tag{40}$$

Note that the actuator speed is in terms of the velocity of the point C_i and the time derivate of C_i is:

$$\begin{bmatrix}\dot{C}_{ix}\\ \dot{C}_{iy}\\ \dot{C}_{iz}\end{bmatrix}=\begin{bmatrix}\dot{P}_x\\ \dot{P}_y\\ \dot{P}_z\end{bmatrix}+\frac{d}{dt}\begin{bmatrix}\cos(\phi_i) & \sin(\phi_i) & 0\\ -\sin(\phi_i) & \cos(\phi_i) & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}h_i\\ 0\\ 0\\ 1\end{bmatrix}\quad\text{where } \phi \text{ is constant and}\quad\begin{bmatrix}\dot{C}_{ix}\\ \dot{C}_{iy}\\ \dot{C}_{iz}\end{bmatrix}=\begin{bmatrix}\dot{P}_x\\ \dot{P}_y\\ \dot{P}_z\end{bmatrix}\tag{41}$$

Substituting the above equation in (40) and the expanding the equation, finally the inverse Jacobian of the robot is given by:

$$\begin{bmatrix}\dot{\theta}_1\\ \dot{\theta}_2\\ \dot{\theta}_3\end{bmatrix}=4\begin{bmatrix}D_{1x} & D_{1y} & D_{1z}\\ D_{2x} & D_{2y} & D_{2z}\\ D_{3x} & D_{3y} & D_{3z}\end{bmatrix}\begin{bmatrix}\dot{P}_x\\ \dot{P}_y\\ \dot{P}_z\end{bmatrix} \tag{42}$$

Note that the robot Jacobian in eq. (42) has the advantage that is fully expressed in terms of physical parameters of the robot and is not necessary to solve previously any kinematic model. Terms in eq. (42) are complex and this make not easy to detect singularities by only inspecting the expression. In the real time controller, the condition number of the jacobian is calculated numerically to detect singularities and subsequently the jacobian is used in the visual controller.

3.3 Robotenis inverse dynamical model

Dynamics plays an important role in robot control depending on applications. For a wide number of applications the dynamical model it could be omitted in the control of the robot. On the other hand there are tasks in which dynamical model has to be taken into account. Dynamic model is important when the robot has to interact with heavy loads, when it has to move at high speed (even vibrating), when the robot structure requires including dynamical model into its analysis (for example in wired and flexible robots), when the energy has to be optimized or saved. In our case the dynamical model make possible that the end effector of the robot reaches higher velocities and faster response. The inverse dynamics, (given the trajectory, velocities and accelerations of the end effector) determine the necessary joint forces or torques to reach the end-effector requirements. The direct dynamics, being given the actuators joint forces or torques, determine the trajectory, velocity and acceleration of the end effector. In this work the inverse dynamical is retrofitted to calculate the necessary torque of the actuator to move the end effector to follow a trajectory at some velocity and acceleration. We will show how the inverse dynamics is used in the joint controller of the robot. Robotenis system is a parallel robot inspired in the delta robot, this parallel robot is relatively simple and its inverse dynamics can be obtained by applying the Lagrangian equations of the first type. The Lagrangian equations of the Robotenis system are written in terms of coordinates that are redundant, this makes necessary a set of constraint equations (and them derivates) in order to solve the additional coordinates. Constraint equations can be obtained from the kinematical constraints of the mechanism in order to generate the same number of equations that the coordinates that are unknown (generalized and redundant coordinates). Lagrangian equations of the first type can be expressed:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right)-\frac{\partial L}{\partial q_j}=Q_j+\sum_{i=1}^{k}\lambda_i\frac{\partial \Gamma_i}{\partial q_j} \qquad j=1,\ 2,\ \ldots,\ n \tag{43}$$

Where $\boldsymbol{\Gamma_i}$ is the $\boldsymbol{i}$ constraint equation, $\boldsymbol{\lambda_i}$ is the Lagrangian multiplier, $\boldsymbol{k}$ is the number of constraint equation, $\boldsymbol{n}$ is the number of coordinates (Note that $\boldsymbol{Degrees\ of\ freedom} =$

$\boldsymbol{n-k}$ and in our case DOF = number of actuated joints), $\boldsymbol{Q}$ contains the external applied forces $\widehat{\boldsymbol{Q}}_{\boldsymbol{k}}$ and the actuator torques or forces $\boldsymbol{Q}_{\boldsymbol{n-k}}$ $(\boldsymbol{Q} = [\widehat{\boldsymbol{Q}}_{\boldsymbol{j}}, \boldsymbol{Q}_{\boldsymbol{j}}] = [\boldsymbol{Q}_{\boldsymbol{j=1,2,\ldots,k}}, \boldsymbol{Q}_{\boldsymbol{j=k+1,\ldots,n}}])$. By means of following considerations, the equations in (43) can be arranged in two sets of equations. Consider that the first $\boldsymbol{k}$ equations are associated with the redundant coordinates and the $\boldsymbol{n-k}$ equations are associated with the actuated joint variables, consider that for the inverse dynamics external forces are given or measured. Thus the first set of equations can be arranged as:

$$\sum_{i=1}^{k} \lambda_i \frac{\partial \Gamma_i}{\partial q_j} = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \frac{\partial L}{\partial q_j} - \hat{Q}_j \qquad j=1,\ 2,\ \ldots,\ k \tag{44}$$

Where the right side is known and for each redundant coordinate yields a set of k linear equations that can be solved for the k Lagrangian multipliers $\lambda_{1,\ldots,k}$. Finally the second set of equations uses the k Lagrangian multipliers to find the actuator forces or torques. Second set of equations can be grouped in:

$$\tilde{Q}_j = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right) - \frac{\partial L}{\partial q_j} - \sum_{i=1}^{k} \lambda_i \frac{\partial \Gamma_i}{\partial q_j} \qquad j=k+1,\ \ldots,\ n \tag{45}$$

Applying the above to the Robotenis system, we have that $\boldsymbol{\theta_{11}}$, $\boldsymbol{\theta_{12}}$ and $\boldsymbol{\theta_{13}}$ can define the full system and can be chosen as generalized coordinates moreover to simplify the Lagrange expression and to solve the Lagrangian by means of Lagrange multipliers we choose 3 additional redundant coordinates $\boldsymbol{P_x}$, $\boldsymbol{P_y}$ and $\boldsymbol{P_z}$. Thus the generalized coordinates are: $\boldsymbol{P_x}, \boldsymbol{P_y}, \boldsymbol{P_z}, \boldsymbol{\theta_{11}}, \boldsymbol{\theta_{12}}$ and $\boldsymbol{\theta_{13}}$. External forces and position, velocity and acceleration of the end effector (mobile platform) are known, thus the six variables are: the three Lagrangian multipliers (they correspond to the three constraint equations) and the three actuators torque. Three constraint equations are obtained from the eq. (10) when points $\boldsymbol{C_{ixyz}}$ are substituted by $\boldsymbol{P_{xyz}}$ by means of eq. (18).

$$\Gamma_i = \left(P_x + (h_i - H_i)c(\phi_i) - a\,c(\phi_i)c(\theta_i)\right)^2 + \left(P_y + (h_i - H_i)s(\phi_i) - a\,s(\phi_i)c(\theta_i)\right)^2 + \left(P_z - a\,s(\theta_i)\right)^2 - b^2 = 0 \tag{46}$$

In the above equation $i = 1, 2, 3$ and to simplify considers that $\theta_{1i} = \ \theta_i$ (this angles are the actuated joint angles) and that $H = H_i$, $h = h_i$; $i = 1, 2, 3$. The Lagrangian equation is obtained from the kinetics and potential energy, thus some considerations are done to simplify the analysis. m_a is the half of the mass of the input link and is supposed to be concentrated at two points (A and B), I is the axial moment of inertia of the input shaft (and the half of the input link), m_b is the half of the mass of the second link (thus m_b is supposed that is concentrated in two points, in the point B and in the point C), m_p is the mass of the end effector and is supposed being concentrated at the point P_{xyz}. Regarding that the translational kinetic energy of a rigid body is: $K_t = \frac{mv^2}{2}$ and if the rigid body is rotating around its center of mass the kinetic energy is: $K_r = \frac{I\omega^2}{2}$, where v is the translational velocity, m is the mass of the body in the center of mass, I is the moment of inertia and ω is the body's angular velocity. Thus the total kinetic energy of the robot is (mobile platform, 3 input links and 3 input shafts, and 3 parallelogram links):

$$K=\frac{1}{2}\left[m_p\left(\dot{p}_x^2+\dot{p}_y^2+\dot{p}_z^2\right)+\left(m_a a^2+I\right)\left(\dot{\theta}_1^2+\dot{\theta}_2^2+\dot{\theta}_3^2\right)+m_b a^2\left(\dot{\theta}_1^2+\dot{\theta}_2^2+\dot{\theta}_3^2\right)+3m_b\left(\dot{p}_x^2+\dot{p}_y^2+\dot{p}_z^2\right)\right] \tag{47}$$

The Potential energy is energy depends on the elevation of the elements ($V = mgP_z$), m is the mass, g is the constant of gravity and P_z is the s the altitude of the gravitated object. In the robot the potential energy of the platform, the input links and the parallelogram links is:

$$V=g\left[m_p P_z+m_a a\left(s(\theta_1)+s(\theta_2)+s(\theta_3)\right)+m_b\left(3P_z+a\left(s(\theta_1)+s(\theta_2)+s(\theta_3)\right)\right)\right] \tag{48}$$

Therefore the Lagrangian function ($L = K - V$) is:

$$\begin{aligned} L=&\frac{1}{2}\left(m_p+3m_b\right)\left(\dot{p}_x^2+\dot{p}_y^2+\dot{p}_z^2\right)+\frac{1}{2}\left(m_a a^2+I+m_b a^2\right)\left(\dot{\theta}_1^2+\dot{\theta}_2^2+\dot{\theta}_3^2\right)\\ &-g\left(m_p+3m_b\right)P_z-a\,g\left(m_a+m_b\right)\left(s(\theta_1)+s(\theta_2)+s(\theta_3)\right) \end{aligned} \tag{49}$$

Taking the partial derivatives of the Lagrangian with respect to the generalized coordinates, we have.

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{P}_x}\right)=\left(m_p+3m_b\right)\ddot{P}_x \qquad \frac{\partial L}{\partial P_x}=0$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{P}_y}\right)=\left(m_p+3m_b\right)\ddot{P}_y \qquad \frac{\partial L}{\partial P_y}=0$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{P}_z}\right)=\left(m_p+3m_b\right)\ddot{P}_z \qquad \frac{\partial L}{\partial P_x}=-g\left(m_p+3m_b\right)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_1}\right)=\left(m_a a^2+I+m_b a^2\right)\ddot{\theta}_1 \qquad \frac{\partial L}{\partial \theta_1}=-a\,g\left(m_a+m_b\right)c(\theta_1)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_2}\right)=\left(m_a a^2+I+m_b a^2\right)\ddot{\theta}_2 \qquad \frac{\partial L}{\partial \theta_2}=-a\,g\left(m_a+m_b\right)c(\theta_2)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_3}\right)=\left(m_a a^2+I+m_b a^2\right)\ddot{\theta}_3 \qquad \frac{\partial L}{\partial \theta_3}=-a\,g\left(m_a+m_b\right)c(\theta_3)$$

Taking the partial derivatives of the constraint equations (46) with respect to the generalized coordinates, we have.